CLAIMS

1. A compiling apparatus characterized by comprising:

an analysis data generating section for generating, form a source program, analysis data of the source program;

a first executable program generating section for generating a first executable program on the basis of the analysis data;

a profile data generating section for generating profile data on the basis of the first executable program; and

a second executable program generating section for generating a second executable program on the basis of the analysis data and the profile data.

2. A compiling apparatus according to claim 1, characterized by further comprising an analysis data storage section for storing the analysis data generated by the analysis data generating section.

3. A compiling apparatus according to claim 1, characterized by further comprising a profile data storage section for storing the profile data generated by the profile data generating section.

4. A compiling apparatus according to claim 1, characterized in that the first executable program generating section and the second executable program generating section constitute one executable program generating section.

5. A compiling apparatus according to claim 1, characterized in that the first executable program generating section generates an executable program that can

be executed in a target computer.

6. A compiling apparatus according to claim 1, characterized in that the first executable program generating section generates a program of an indirect execution format.

7. A compiling apparatus according to claim 6, characterized in that the first executable program generating section is one for generating an intermediate program of an indirect execution format, which can be executed regardless of the type of the computer that executes the program.

8. A compiling apparatus according to claim 1, characterized by further comprising a compile designation data receiving section for receiving compile designation data from an operator and characterized in that the second executable program generating section generates an optimal program for a target computer, on the basis of the compile designation data input via the compile designation data receiving section, the analysis data and the profile data.

9. A compiling apparatus according to claim 1, characterized by further comprising a compile designation data receiving section for receiving compile designation data from an operator and a compile designation data storage section for storing the compile designation data input via the compile designation data receiving section, and characterized in that the second executable program generating section generates an optimal program for a target computer, on the basis of the compile designation data stored in the compile designation data storage section, the analysis

data and the profile data.

10. A compiling apparatus according to claim 1, characterized by further comprising a compile designation data receiving section for receiving compile designation data from a program that assists a compiling process, and characterized in that the second executable program generating section generates an optimal program for a target computer, on the basis of the compile designation data input via the compile designation data receiving section, the analysis data and the profile data.

11. A compiling apparatus according to claim 1, characterized by further comprising a compile designation data receiving section for receiving compile designation data from a program that assists a compiling process and a compile designation data storage section for storing the compile designation data received via the compile designation data receiving section, and characterized in that the second executable program generating section generates an optimal program for a target computer, on the basis of the compile designation data stored in the compile designation data storage section, the analysis data and the profile data.

12. A compiling method characterized by comprising:

an analysis data generating step of generating, form a source program, analysis data of the source program;

a first executable program generating step of generating a first executable program on the basis of the analysis data;

a profile data generating step of generating profile data on the basis of the first executable program; and

a second executable program generating step of generating section for generating a second executable program on the basis of the analysis data and the profile data.

13. A compiling method according to claim 12, characterized by further comprising an analysis data storing step of storing the analysis data generated by the analysis data generating section, in an analysis data storage section.

14. A compiling method according to claim 12, characterized by further comprising a profile data storing step of storing the profile data generated by the profile data generating section, in a profile data storage section.

15. A compiling method according to claim 12, characterized in that the first executable program generating generates an executable program that can be executed in a target computer.

16. A compiling method according to claim 12, characterized in that the first executable program generating step generates a program of an indirect execution format.

17. A compiling method according to claim 16, characterized in that the first executable program generating step generates an intermediate program of an indirect execution format, which can be executed regardless of the type of the computer that executes the program.

18. A compiling method according to claim 12, characterized by further

comprising a compile designation data receiving step of receiving compile designation data from an operator and characterized in that an optimal program for a target computer is generated in the second executable program generating step, on the basis of the compile designation data input via the compile designation data receiving section, the analysis data and the profile data.

19. A compiling method according to claim 12, characterized by further comprising a compile designation data receiving step of receiving compile designation data from an operator and a compile and storing the same in a compile designation data storing section, and characterized in that an optimal program for a target computer is generated in the second executable program generating step, on the basis of the compile designation data stored in the compile designation data storage section, the analysis data and the profile data.

20. A method according to claim 12, characterized by further comprising a compile designation data receiving step of receiving compile designation data from a program that assists a compiling process, and characterized in that an optimal program for a target computer is generated in the second executable program generating step, on the basis of the compile designation data input via the compile designation data receiving section, the analysis data and the profile data.

21. A compiling method according to claim 12, characterized by further comprising a compile designation data receiving step of receiving compile designation data from a program that assists a compiling process and storing the same in a compile

designation data storage section, and characterized in that an optimal program for a target computer is generated in the second executable program generating step, on the basis of the compile designation data stored in the compile designation data storage section, the analysis data and the profile data.